

CLAIMS

What is claimed is:

- 15 Sub
C1 A3
2
1. A processor comprising:
 - 3 a first instruction set engine;
 - 4 a second instruction set engine;
 - 5 a mode identifier;
 - 6 a plurality of floating-point registers shared by the first instruction set engine and the
 - 7 second instruction set engine; and
 - 8 a floating-point unit coupled to the floating-point registers, the floating-point unitprocessing an input responsive to the mode identifier to produce an output.
 - 1 2. The processor of Claim 1 wherein the mode identifier is one of a plurality of bits
 - 2 in a processor status register.
 - 1 3. The processor of Claim 1 wherein the floating-point unit comprises:
 - 2 pre-processing hardware to detect if a token exists in the input;
 - 3 an arithmetic unit responsive to the input and the mode identifier; and
 - 4 post-processing hardware to perform a token specific operation if a token exists in the
 - 5 input.
 - 1 4. The processor of Claim 1 wherein the input includes data stored in at least one of
 - 2 the floating-point registers.

Cont
CI

1 5. The processor of Claim 1 wherein the input may contain a token, wherein the
2 floating-point registers are 82 bits wide, and wherein the token being an 82 bit processor known
3 value.

1 6. The processor of Claim 3 wherein the token represents a "not a thing value"
2 (NaTVal) that defines an unsuccessful speculative load request

1 7. The processor of Claim 1 wherein the floating point registers each comprise:
2 a sign bit,
3 an exponent; and
4 a significand.

1 8. The processor of Claim 1 wherein the mode identifier indicates whether the
2 processor is in a first mode or a second mode.

1 9. The processor of Claim 1 wherein the mode identifier indicates whether the
2 processor is in a 32 bit word instruction set architecture mode (ISA) or a 64 bit word ISA mode.

1 10. A method in a processor comprising:
2 fetching an input from at least one of a plurality of floating-point registers;
3 detecting whether the input includes a token;
4 if the token is detected in the input, checking what mode the processor is in;
5 if the processor is in a first mode, processing the input to render an arithmetic result;
6 if the processor is in a second mode, performing a token specific operation; and
7 producing an output.

1 Sub
2 11. The method of Claim 10 wherein the input is comprised of at least one operand
3 and at least operator; wherein detecting comprises examining the at least one operand to
4 determine whether any of the operands correspond to the token; and wherein checking comprises
5 examining a mode identifier to determine whether the processor is in the first mode or the second
mode.

1 12. The method of Claim 10 wherein processing comprises executing at least one
2 operation on the at least one operand according to the at least one operator to achieve a result.

1 13. The method of Claim 10 wherein performing comprises propagating the token;
2 and wherein producing output comprises setting the output to be the token.

1 14. The method of Claim 10 wherein the token represents a "not a thing value"
2 (NaTVal) that defines an unsuccessful speculative load request.

1 15. The method of Claim 10 wherein checking comprises checking a mode identifier.

1 16. The method of Claim 10 wherein checking comprises checking a mode identifier
2 bit in a processor status register.

1 17. The method of Claim 11 wherein the first mode is a 32 bit word ISA mode and
2 the second mode is a 64 bit word ISA mode.

1 18. A multi-mode processor comprising:

2 a plurality of instruction set engines;

3 a mode identifier;

4 a plurality of floating-point registers shared by the instruction set engines; and

Cont
C1
5 a plurality of floating-point units coupled to the floating-point registers, the floating-point
6 units processing an input responsive to the mode identifier.

1 19. A method in a multi-mode processor comprising:
2 fetching an input from at least one of a plurality of floating-point registers;
3 detecting whether the input includes at least one token of a plurality of tokens;
4 if at least one token is detected in the input, checking what mode the processor is in;
5 processing the input to render an arithmetic result when the processor is in at least a first
6 mode of a plurality of modes; and
7 performing a token specific operation when the processor is in at least a second mode of a
8 plurality of modes.